Brewer Protocol

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 1 of 1 returned.

☐ 1. Document ID: US 6876656 B2

L4: Entry 1 of 1 File: USPT Apr 5, 2005

DOCUMENT-IDENTIFIER: US 6876656 B2
TITLE: Switch assisted frame aliasing for storage virtualization

Brief Summary Text (4):
The way a prior art network such as a Fibre Channel Network works to read and write data between client devices and storage devices was as follows. Referring to FIG. 1, a client 10 which wishes to retrieve data from the storage manager would address a Fibre Channel (FC) frame to the server 20 (all the prior art transport protocols and primitives will not be described as they not are part of the invention other than as the basic platform on which the invention sits). This frame contains a SCSI command requesting the desired data. The frame will have a header that contains address information and a payload which contains a SCSI command. The address (PA) of storage client will be the source address, and the address of the server will be the destination address. The header of each frame also contains two exchange IDs, one for the originator and one for the responder, that serves to identify all the frames that belong to this particular read or write transaction. If the same client has, for example, two read or write transaction outstanding, all the frames transmitted from that originator client pertaining to either of those transactions will have the same source and destination address, but all the frames pertaining to the first transaction will have a first originator exchange ID, and all the frames pertaining to the second transaction will have a second, different originator exchange ID. There are also flags to indicate the type of data contained in the payload section of the FC frame such as: a command to do a read or write, a transfer ready message or the requested data itself.

Detailed Description Text (2):
Referring to FIGS. 2A through 2B, there is shown a table giving the sequence of events that must occur to implement the process of one species within the invention in carrying out a read transaction and redirecting all frames. Basically, the invention can be implemented by modifying the software in the storage manager 20 and switch 14 in the prior art structure of FIG. 1 to carry out the sequence of events listed in the tables of FIGS. 2A through 2B. The term "switch" as used in the claims should be understood as including not only single packet or frame switches but also combinations of physically separate switches all coupled to form a network and having a protocol to exchange information with each other to implement the switching function.

2/10/05

Detailed Description Text (14):
In the particular example given in FIG. 5, for a read request, two redirection commands are utilized to cover embodiments where there is a protocol requiring the host to issue a transer ready frame when it is ready to receive the read data. However, in existing FC networks, the host does not issue the read command until it is ready to receive the read data. Thus, in the preferred embodiment, only one

redirection command is needed on a read request, and that redirection command works only on responder frames traveling from the device to the host. This second redirection command is represented by arrow 56 in FIG. 5, and is sent to a redirection process 74 in port 60 coupled to storage device 18, and is stored in lookup table 54. Thus, in the preferred embodiment, the redirection command 58 is not needed. In embodiments where it is needed, it is sent to redirection process 76 of port 62 coupled to the host and stored in lookup table 52.

Detailed Description Text (130):
However, the invention may also be used in packet based LAN and WAN environments where there is a storage manager or other bottleneck device in which it is advantageous to divert packets around. In these environments, the protocols are different, so any method of the storage manager determining when the transaction is completed given the particular protocol will suffice. In these alternative embodiments, the storage manager determines the transaction has been completed by watching the network traffic for a message that the transaction is done or by a timeout on receiving packets having the particular exchange ID or by any other way supported by the protocol. When the storage manager determines that the transaction is done, it finds the appropriate redirection command or commands and sends purge commands to the switch or router that is doing the redirection.

Detailed Description Text (131):
The switch 14 in each of FIGS. 5 through 9 should be understood in this example and for purposes of interpreting the term "switch" in the claims as being either a single switch or multiple switches which are all connected together in a network and which can communicate by an information protocol to implement the switching and redirection function. In a typical situation with multiple switches in the same network, each switch stores configuration data as to only the local devices to which it is connected. This switch is coupled to the other switches in the network by network data paths. The switches all communicate with each other to route packets and frames properly. Thus, in the example of FIG. 7, suppose that switch 14 was really two different switches, one on the West Coast coupled to some client and storage devices and a storage manager and coupled to another switch on the East Coast by a T.sub.1 or other WAN connection. Suppose client 10 is on the West Coast and storage device 18 is on the East Coast. When a write transaction is initiated by client 10 that involves writing data to storage device 18, the storage manager issues redirection commands 58A and 56A to the West Coast switch. The West Coast switch looks at its configuration data 81 and does not find storage device 18. It then carries out a communication protocol with the East Coast switch and any other switches in the network if there are more to find out which switch is coupled to storage device 18. The East Coast switch replies that it is so connected. The West Coast switch then sends redirection command 56B to the East Coast switch which looks up storage device 18 in its lookup table and routes redirection command 56B to redirection process 74 in port 60 which stores the old and new address data in lookup table 54. The redirection process 74, if necessary, alters the new address data or appends data to it to indicate that frames or packets to be redirected in this write transaction need to be sent over the network path to the West Coast switch for further routing.

Detailed Description Text (147):
Test 108 monitors if the transaction is done according to whatever network protocol is in existence. If the transaction is not done, processing returns again to step 108 to wait for the transaction to complete. The flowchart of FIG. 10 applies to each exchange, and multiple instances of the process of FIG. 10 may be occurring in the storage manager if multiple exchanges are occurring simultaneously.

Detailed Description Text (149):
Referring to FIG. 11, which is comprised of FIGS. 11A and 11B, there is shown a flowchart of the processing of the redirection process in each port or the central redirection process to process incoming frames. The flowchart shows only the

processing at the logical level of redirection and not any physical layer processing to carry out the physical layer or other protocols that happen regardless of whether a frame is or is not redirected. Each function represented by a step in FIG. 11 can be performed either by a programmed microprocessor, a gate array, an ASIC or a field programmable gate array or conventional switch circuitry such as the routing engine. Further, both FIGS. 10 and 11 are merely exemplary processing flows and the exact process flow of other species within the genus does not to be the same so long as the same functions are achieved in an order which causes the end results achieved by FIGS. 10 and 11 to occur.

Detailed Description Text (158):
Referring to FIG. 12, there is shown a block diagram of the pertinent hardware and software modules of a typical storage manager that can implement the teachings of the invention. Basically, a programmed microprocessor, gate array, field programmable gate array or ASIC can be structured to perform the conventional and redirection functionality of the storage manager so long as the circuitry chosen can handle the bandwidth involved. In FIG. 12, a programmed CPU 134 has been chosen to represent all these possibilities but in the claims, all these possible structures are referred to as a "computer programmed to or circuitry structured to make the decision to redirect . . . " The CPU is programmed with software 135 that carries out conventional storage manager processing as well as the redirection processing. Basically, these conventional and redirection applications interact with the storage manager operating system to carry out the redirection processing and other conventional processing symbolized by FIGS. 10 and 13. The CPU is coupled to a input/output bus 136 for communication with a memory 138 and one or more port adapters of which port adapter 140 is typical. In some embodiments of the storage manager, a cache memory 137 is used and a cache algorithm 139 controls the computer 134 to manage the cache conventionally and to cooperate with the redirection process in the manner described elsewhere herein. The function of the port adapter 140 is to implement the physical layer protocol and any other protocols necessary to communicate with the switch 14.

Term	Documents
PROTOCOL	128146
PROTOCOLS	83686
(1 AND PROTOCOL).USPT.	1
(L1 AND PROTOCOL).USPT.	1

Freeform Search

Database: US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: L1 and (stateful with protocol$)

Display: 10 **Documents in Display Format:** KWIC **Starting with Number** 1

Generate: ○ **Hit List** ◉ **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Monday, August 01, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=ADJ			
L2	L1 and (stateful with protocol$)	29	L2
L1	709/$.ccls.	18389	L1

END OF SEARCH HISTORY

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L2: Entry 7 of 29 File: USPT Nov 30, 2004

DOCUMENT-IDENTIFIER: US 6826613 B1
TITLE: Virtually addressing storage devices through a switch

Detailed Description Text (56):
The NFS and CIFS file structures supported by the present invention are industry standard file structures. NFS is a stateless system that can use TCP or UDP. The default is to use TCP and to fall back to UDP if TCP is unavailable. CIFS is a stateful protocol that can use UDP or TCP. In practice, however, no UDP implementation is known to exist. It is anticipated that TCP-based protocols will provide better performance, especially over WAN connections. TCP has the benefit that if the client fails, the server can immediately release client resources and vice-versa if the server fails. TCP performance features (slow start, queuing priority, etc.) can be enhanced utilizing a Border Gateway Protocol (BGP)-style usage of TCP or other commonly used approaches. CEFS is a much more complex protocol than NFS, which supports extended attributes and passing of device input/output control (IOCTL) information.

Current US Original Classification (1):
709/227

Current US Cross Reference Classification (1):
709/202

Current US Cross Reference Classification (2):
709/203

Current US Cross Reference Classification (3):
709/213

Current US Cross Reference Classification (4):
709/215

Previous Doc Next Doc Go to Doc#

US006826613B1

(12) **United States Patent**
Wang et al.

(10) **Patent No.: US 6,826,613 B1**
(45) **Date of Patent: Nov. 30, 2004**

(54) **VIRTUALLY ADDRESSING STORAGE DEVICES THROUGH A SWITCH**

(75) Inventors: **Peter S. S. Wang**, Cupertino, CA (US); **David C. Lee**, San Jose, CA (US); **Anne G. O'Connell**, Cork (IE)

(73) Assignee: **3Com Corporation**, Marlborough, MA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/525,966**

(22) Filed: **Mar. 15, 2000**

(51) **Int. Cl.**[7] **G06F 15/16**

(52) **U.S. Cl.** **709/227**; 709/202; 709/203; 709/213; 709/215

(58) **Field of Search** 709/200–203, 709/226–229, 238, 242, 213–215; 370/400–404, 409

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,522,045	A *	5/1996	Sandberg	709/215
5,819,036	A *	10/1998	Adams et al.	709/203
6,081,833	A *	6/2000	Okamoto et al.	709/213
6,085,238	A *	7/2000	Yuasa et al.	370/409
6,088,728	A *	7/2000	Bellemore et al.	709/227
6,252,878	B1 *	6/2001	Locklear, Jr. et al.	370/401
6,421,711	B1 *	7/2002	Blumenau et al.	709/213
6,601,101	B1 *	7/2003	Lee et al.	709/227

OTHER PUBLICATIONS

"Active Disks—Remote Execution for Network–Attached Storage" Article can be found at: pdl.cs.cmu.edu:80/Active/index.html.

"Extreme NASD," Article printed out on Aug. 4, 1999. The article was found at pdl.cs.cmu.edu:80/extreme/, but it is no longer available on the web.

"High–bandwidth, Low–latency, and Scalable Storage Systems," Article can be found at: pdl.cmu.edu/NASD/index.html.

McKusick et al. "The Design and Implementation of the 4.4 BSD Operating System", Addison–Wesley, 1996, Chapter 6 pp. 193–201.

McKusick et al. "The Design and Implementation of the 4.4 BSD Operating System", Addison–Wesley, 1996, Chapter 7 pp. 241–263.

McKusick et al. "The Design and Implementation of the 4.4 BSD Operating System", Addison–Wesley, 1996, Chapter 9 pp. 311–336.

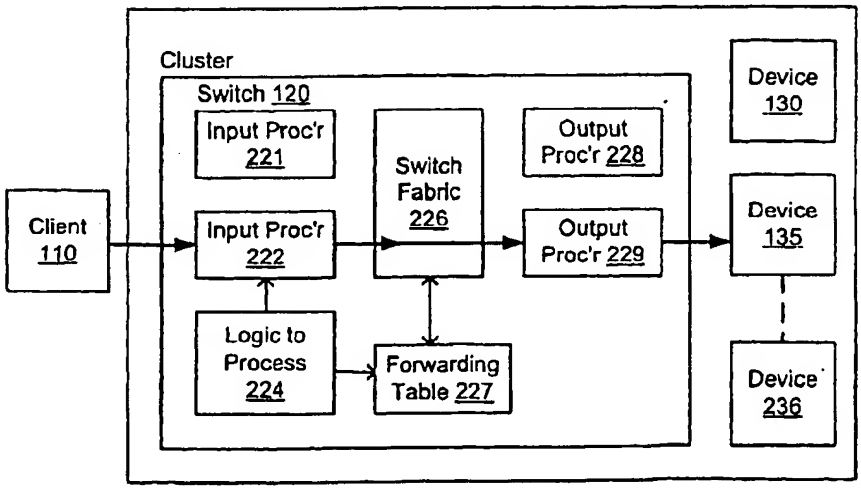
* cited by examiner

Primary Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Ernest J. Beffel, Jr.; Haynes Beffel & Wolfeld LLP

(57) **ABSTRACT**

The present invention relates to transparent access to network attached storage devices, configured to any of several protocols, such as SCSI over IP, NAS or NASD. In particular, the present invention provides for using a switch to transparently aggregate storage devices. The switch appears as a virtual storage device. It responds to requests to initiate file sessions and selects one of a plurality of storage devices to participate in the file session. A file session can be handed off to a different storage device. Both the setup and handoff are transparent to the client and its TCP/IP client. The present invention may be practiced either as a method or device. It may provide a virtual storage device or it may aggregate storage devices already attached to a network.

58 Claims, 8 Drawing Sheets



L2 + L3 both network, network internet in Brenza

Freeform Search

Database:
- US Pre-Grant Publication Full-Text Database
- **US Patents Full-Text Database**
- US OCR Full-Text Database
- EPO Abstracts Database
- JPO Abstracts Database
- Derwent World Patents Index
- IBM Technical Disclosure Bulletins

Term: L1 and (network$ or internet)

Display: 10 **Documents in Display Format:** KWIC **Starting with Number** 1

Generate: ○ **Hit List** ◉ **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Monday, August 01, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=ADJ			
L3	L1 and (network$ or internet)	1	L3
L2	L1 and session$	1	L2
L1	6876656.pn.	1	L1

END OF SEARCH HISTORY

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 1 of 1 returned.

☐ 1. Document ID: US 6876656 B2

L2: Entry 1 of 1 File: USPT Apr 5, 2005

DOCUMENT-IDENTIFIER: US 6876656 B2
TITLE: Switch assisted frame aliasing for storage virtualization

Brief Summary Text (16):
Overview of how the Director Functions in Http Session Redirector Mode: 1. A client web browser tries to retrieve URL http://www.sleet.com. 2. The Internet DNS system maps this name to the Director virtual IP address 10.0.0.4. 3. The Director listens for HTTP connections to IP address 10.0.0.4. 4. The client web browser connects to IP address 10.0.0.4. 5. The Director performs a look up for the host name associated with the address 10.0.0.4. 6. The Director performs a look up for the IP addresses associated with the host name www-servers.sleet.com. This results in the normal Director sorting of addresses using all of the metrics configured for this host name. 7. The Director then constructs the new URL using the IP address of the discovered "best" web server (for example, http://12.0.0.2) appended with the rest of the original URL, and sends the web client the code "302 Temporarily Moved," specifying the new URL location. If the URL originally requested had been: http://www.sleet.com/Weather/index.html Then the new URL would be: http://12.0.0.2/Weather/index.html 8. The client web browser receives the temporary relocation code and transparently connects to the web server at the specified URL.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWMC | Draw. D

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Term	Documents
SESSION$	0
SESSION	32712
SESSIONA	6
SESSIONABLEBO	1
SESSIONABNORMAL	2
SESSIONAGENT	3
SESSIONAID	1
SESSIONAL	27

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 1 of 1 returned.

☐ 1. Document ID: US 6876656 B2

L3: Entry 1 of 1 File: USPT Apr 5, 2005

DOCUMENT-IDENTIFIER: US 6876656 B2
TITLE: Switch assisted frame aliasing for storage virtualization

Abstract Text (1):
An apparatus and process for relabelling and redirecting at least some of the read transaction data frames and the write transaction write data and transfer ready frames in a network so as to bypass a storage manager and pass directly between the client and a storage device via a switch. This eliminates the storage manager as a bottleneck. Some embodiments redirect every read and write transaction, and others redirect only large transactions, or only ones not stored in cache or when latency gets too high. Redirection is accomplished by transmission from the storage manager to a switch redirection commands that contain old and new address data. When a frame to be redirected comes in, its address is compared to the old address data. If there is a match, the new address data is substituted and the frame is passed to a conventional routing process to be routed so as to bypass the storage manager.

Brief Summary Text (2):
The invention finds application in data processing systems such as storage area networks which have at least: (1) an interconnect network which transports data in packets; (2) a storage client or clients; (3) a storage server; and (4) storage devices. In such networks the storage server manages a large number of storage devices to retrieve and store data for various storage clients. The storage clients are not directly connected to the storage devices, and request data stored on the storage devices by making requests to the storage server. The storage server then makes a request to the storage devices. The network, comprised of physical transmission medium and various devices such as hubs, switches, routers etc. provides for the actual transport of data between the clients and the storage manager in the storage server and the transport of data between the storage server and the storage devices. The network also provides a data path between the storage clients and the storage devices. Any connections between the storage clients and the storage devices are not used, because the storage server needs to be solely responsible for the organization of data on the storage devices.

Brief Summary Text (3):
FIG. 1 shows a typical prior art network configuration implemented with a switch. Storage clients 10 and 12 are coupled to two different ports of switch 14. The switch is also coupled to storage devices 16 and 18 through two different ports. A storage server 20 implementing a storage manager process has an input 22 coupled to one port and an output 24 coupled to another port. The switch allows each port to be coupled to any other port and allows multiple simultaneous connections. Thus, data paths between the clients and the server and between the server and the storage devices can be set up through the switch. In addition, data paths can be set up between the storage clients and the storage devices through the switch, but

the clients have no use for this since the clients recognize only the server/storage manager as a storage provider even though the actual data is stored on the storeage devices.

Brief Summary Text (4):
The way a prior art network such as a Fibre Channel Network works to read and write data between client devices and storage devices was as follows. Referring to FIG. 1, a client 10 which wishes to retrieve data from the storage manager would address a Fibre Channel (FC) frame to the server 20 (all the prior art transport protocols and primitives will not be described as they not are part of the invention other than as the basic platform on which the invention sits). This frame contains a SCSI command requesting the desired data. The frame will have a header that contains address information and a payload which contains a SCSI command. The address (PA) of storage client will be the source address, and the address of the server will be the destination address. The header of each frame also contains two exchange IDs, one for the originator and one for the responder, that serves to identify all the frames that belong to this particular read or write transaction. If the same client has, for example, two read or write transaction outstanding, all the frames transmitted from that originator client pertaining to either of those transactions will have the same source and destination address, but all the frames pertaining to the first transaction will have a first originator exchange ID, and all the frames pertaining to the second transaction will have a second, different originator exchange ID. There are also flags to indicate the type of data contained in the payload section of the FC frame such as: a command to do a read or write, a transfer ready message or the requested data itself.

Brief Summary Text (5):
The sequence of events for write and read data transfer operations in a prior art network like that shown in FIG. 1 has the exchanges defined in Table 1 below. In the prior art data transfers, the originator would be a storage client and the responder would be the storage manager 20 for both read and write transactions.

Brief Summary Text (7):
The way this sequence of events works in the prior art networks is that the client 10 sends a SCSI command to, for example, read data to the storage manager. This request will be transmitted to the storage manager through the switch by encapsulating the SCSI command in a FC frame or other packet, as represented by line 1 of Table 1. The read command will request reading of data and specify the desired data by, for example, specifying that the desired data resides on SCSI Logical Unit 1, starting at logical block 75 and extends for 200 logical blocks. This read request will have as its destination address, the address of the storage manager server 20 (hereafter the storage manager or server), and will have an originator exchange ID assigned by the client 10 for this transaction, and the responder exchange ID will be blank.

Brief Summary Text (15):
There is an existing, related process called Web Director available commercially from Cisco that performs redirecting of web requests sent to a first server to a second server in order to offload work to the other servers. When a web request is received at a first server, it is mapped to a second server, and a message is sent back to the client telling it that the web server has been temporarily moved. The web client then transparently connects through the internet to the second server and communicates directly with it. An overview of this process is as follows:

Brief Summary Text (16):
Overview of how the Director Functions in Http Session Redirector Mode: 1. A client web browser tries to retrieve URL http://www.sleet.com. 2. The Internet DNS system maps this name to the Director virtual IP address 10.0.0.4. 3. The Director listens for HTTP connections to IP address 10.0.0.4. 4. The client web browser connects to IP address 10.0.0.4. 5. The Director performs a look up for the host name